

Detekce vodorovného dopravního značení v obrazech

Road Markings Detection in Images

Zadání bakalářské práce

Student:

Jan Štěpán

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Detekce vodorovného dopravního značení v obrazech
Road Markings Detection in Images

Zásady pro vypracování:

Inteligentní asistenční systémy v dopravě se stávají důležitým vybavením moderních vozidel. Jedním z takových asistenčních systému je rozpoznávání vodorovných dopravních značek, tj. značek umístěných přímo na vozovce. Cílem této bakalářské práce je vytvořit metodu pro detekci vodorovných dopravních značek na základě analýzy obrazu. Rozsah přípustných dopravních značek konzultujte s vedoucím práce.

Ve své práci proveďte:

1. Popište zadaný problém.
2. Analyzujte řešení a popište potřebnou teorii.
3. Proveďte vlastní implementaci řešení a její testování.
4. Zhodnoťte dosažené výsledky.

Seznam doporučené odborné literatury:

Podle pokynů vedoucího bakalářské práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

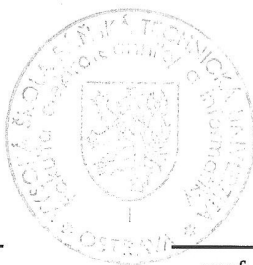
Vedoucí bakalářské práce: **Ing. Michael Holuša**

Datum zadání: 16.11.2012

Datum odevzdání: 07.05.2013



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2013

.....Jan Štejsník.....

Děkuji Ing. Michaeli Holušovi za cenné rady, připomínky a za čas, který mi věnoval při psaní mé bakalářské práce. Poděkování náleží i mé rodině, která mě plně podporovala po celou dobu studia.

Abstrakt

Cílem této bakalářské práce bylo zaměřit se na detekci dopravního značení a rozpoznání o jakou dopravní značku se jedná. Tato bakalářská práce je zaměřena na vodorovné dopravní značky. Vychází z faktu, že tyto dopravní značky mají obvykle bílou barvu nebo žlutou barvu a to v případě přechodného dopravního značení. Bakalářská práce je rozdělena do dvou částí a to na teoretickou a vlastní implementaci.

V teoretické části popisuje obecné zpracování obrazů včetně grafického znázornění daných problémů. Ve vlastní implementaci pak popisuje jednotlivé kroky při vytváření aplikace na detekci vodorovného dopravního značení a to na základě pořízeného videozáznamu. Výsledky fungování této aplikace jsou shrnuty a popsány v závěrečné kapitole.

Klíčová slova: detekce, transformace, dopravní značky, prahování obrazu

Abstract

The subject of the Bachelor thesis is the detection and recognition of the traffic signs. The thesis specifically focuses on road markings. It is based on the fact that road markings are usually white or yellow and are used for temporary traffic marking. The thesis is divided into two parts, theory and implementation.

The theoretical part describes the general image processing, including the graphical representation of the problem. The implementation part uses recorded video and describes the steps in creating applications for the detection of road markings. The results are summarized and described in the final chapter.

Keywords: detection, transformation, road marking, thresholding

Seznam použitých zkratk a symbolů

apod.	– a podobně
atd.	– a tak dále
cca	– circa (cirka)
FPS	– snímků za sekundu (z anglického Frames per second)
HSV	– Hue, Saturation, Value
km/h	– kilometr za hodinu
MS	– Microsoft
ms	– milisekunda
např.	– například
OpenCV	– Open Source Computer Vision
RGB	– Red, Green, Blue
SURF	– Speeded Up Robust Features
SVM	– Support Vector Machines
tj.	– to jest
tzv.	– takzvaný
vs.	– versus

Obsah

1	Úvod	3
2	Obecné informace	5
2.1	Dopravní značky	5
2.2	Počítačové vidění	6
3	Zpracování obrazu	7
3.1	Barevné modely	7
3.2	Snímání a digitalizace obrazu	11
3.3	Předzpracování obrazu	11
3.4	Segmentace obrazu	13
3.5	Popis objektu	17
3.6	Klasifikace	18
4	Vlastní implementace řešení	20
4.1	Získání obrazu	20
4.2	Transformace obrazu	20
4.3	Odstranění šumu	22
4.4	Segmentace dopravních značek	22
4.5	Oblast zájmu a zmenšení obrazu	24
4.6	Vytvoření šablon dopravních značek	25
4.7	Porovnání objektu na obrázku se šablonami	26
4.8	Určení dané dopravní značky	27
5	Zhodnocení dosažených výsledků	29
6	Závěr	32
7	Reference	33

Seznam obrázků

1	Dopravní značky, které je program schopen určit	3
2	Aditivní míchání barev	8
3	Grafické znázornění HSV	8
4	Porovnání barevného obrazu a obrazu v odstínech šedi	10
5	Černobílý obraz	10
6	Konvoluce- rozmazání obrazu	12
7	Geometrická transformace v rovině	13
8	Segmentace- prahováním	15
9	Thresholding	16
10	Threshold Binary	16
11	Jednotlivé kroky zpracování obrazu	20
12	Transformace obrazu	21
13	Segmentace obrazu prahováním	23
14	Použití dilatace a eroze	24
15	Vybrané oblasti zájmu	25
16	Šablona dopravních značek	26
17	Porovnání dopravní značky se šablonou	28
18	Výsledek po detekci a určení dopravní značky	28
19	Ukázka chybné detekce	30

1 Úvod

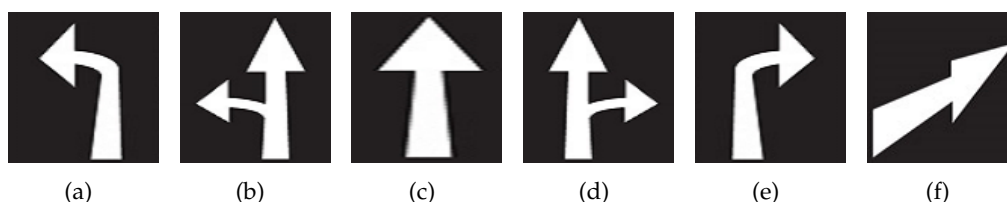
V současné době jsou na automobilový trh uváděny vozy, které mají spoustu bezpečnostních prvků. Za pomoci laserů, čidel a kamer jsou tyto vozy schopny samy rozpoznat dopravní značení, překážky na silnici nebo i mikrosпánek řidiče. Vědecké týmy z oblasti automobilového průmyslu stále pracují na variantách bezpečného silničního provozu a zároveň testují automobily, které jsou plně řízené počítačem.

Například inteligentní kamery pro počítačové vidění ke své funkci nevyžadují PC. Zpracování obrazu probíhá přímo v kameře a PC se používá pouze při programování nebo nastavení. Tyto kamery jsou vyráběny v různých variantách. Nejjednodušší a zároveň nejlevnější z nich, se nazývají inteligentní fotobuňky.

V současnosti dokážou nejlépe vybavené inteligentní kamery snímat v rozlišení až 5 megapixelů a mají nejmodernější nástroje počítačového vidění a zároveň nabízejí různé možnosti komunikace.

Cílem této bakalářské práce bylo zaměřit se na detekci vodorovného dopravního značení a rozpoznání o jakou dopravní značku se jedná. Zpracování obrazu by mělo probíhat v reálném čase.

Tato bakalářská práce je zaměřena na vodorovné dopravní značky. Značky, které by měl program rozpoznat jsou uvedeny na obrázku 1.



Obrázek 1: Dopravní značky, které je program schopen určit[10]

V teoretické části se zaměřím na popis obecného zpracování obrazu a jednotlivé barevné modely, které se při něm využívají. Ve vlastní implementaci se zaměřím na jednotlivé kroky, jak program video zpracuje a následně na něm detekuje vodorovné dopravní značky.

Vycházím z faktu, že tyto dopravní značky mají bílou barvu (výjimku tvoří přechodné dopravní značení, které je specifikováno v § 64 zákona č. 361/2000 Sb., ve znění pozdějších předpisů).

Program zamýšlím otestovat na videozáznamu, ve kterém se objeví všechny značky, které jsou uvedeny na obrázku 1. Mým záměrem je potřebné videosekvence získat

v Ostravě, kde je dostatek vodorovného dopravního značení a zároveň lze najít různé typy tohoto značení.

Dosažené výsledky zhodnotím v závěrečné kapitole a zároveň získám odpovědi na otázky, s jakou přesností dokáže aplikace dopravní značky rozpoznat, jak dlouho toto zpracování a následné rozpoznání bude trvat a kterou dopravní značku dokáže aplikace nejlépe určit.

2 Obecné informace

Tato kapitola popisuje, co jsou to dopravní značky a jaké existují druhy dopravních značek. Dále kapitola obsahuje definici počítačového vidění a oblasti, ve kterých se s ním lze setkat.

2.1 Dopravní značky

Následující definice dopravního značení je čerpána ze zákona o provozu na pozemních komunikacích. [18]

- *Rozlišují se dopravní značky svislé a vodorovné.*
- *Svislé dopravní značky jsou stálé, proměnné a přenosné. Proměnná svislá dopravní značka je dopravní značka, jejíž činná plocha se může měnit. Přenosnou svislou dopravní značkou se rozumí dopravní značka umístěná na červenobíle pruhovaném sloupku (stojánku) nebo na vozidle.*
- *Vodorovné dopravní značky jsou stálé a přechodné. Vodorovné dopravní značky mohou být doplněny dopravními knoflíky.*
- *Tvary symbolů dopravních značek se nesmějí měnit; to neplatí pro dopravní značky se symboly, které mohou být obráceny, a se symboly, které jsou uvedeny jen jako vzory, a pro svislé dopravní značky proměnné.*
- *Prováděcí právní předpis stanoví význam, užití, provedení a tvary dopravních značek a jejich symbolů.*
- *Dopravní značky, světelné a akustické signály, dopravní zařízení a zařízení pro provozní informace musí svými rozměry, barvami a technickými požadavky odpovídat zvláštním technickým předpisům.*

Vodorovné dopravní značky se užívají samostatně nebo ve spojení se svislými dopravními značkami, popřípadě s dopravními zařízeními, jejichž význam zdůrazňují nebo zpřesňují. Vodorovné dopravní značky jsou vyznačeny barvou nebo jiným srozumitelným způsobem; přechodná změna místní úpravy provozu na pozemních komunikacích je vyznačena žlutou nebo oranžovou barvou.

2.2 Počítačové vidění

Počítačové vidění (z anglického *computer vision*) je oblast výpočetní techniky, pomocí které je možné získat informace ze zachyceného obrazu. Tento obraz může být videozáznam, statický snímek nebo data z lékařských vyšetření.

V knize Computer Vision [7] je definováno počítačové vidění následujícím způsobem:

Definice 2.1 *Cílem počítačového vidění je dělat užitečná rozhodnutí o reálných fyzických objektech a scénách na základě snímání obrazu.*

Pomocí počítačového vidění lze rozpoznávat ručně psané texty, zabírat se analýzou provozu na komunikacích, lze zjišťovat různé překážky na komunikaci, rozpoznávat určité objekty atd. Rovněž lze najít uplatnění na letištích, kde lze pomocí počítačového vidění rozpoznávat obličeje a následně vyhodnotit jestli není daná osoba hledaná.

Zde je uvedeno několik příkladů, kde se lze s počítačovým viděním setkat:

- V průmyslu na výrobních linkách, kde se využívají průmysloví roboti. Ti dokážou porovnávat výrobky a rozhodnout jestli jsou v normě nebo naopak poškozené.
- V mobilních telefonech lze počítačové vidění najít v různých aplikacích, jako jsou například čtečky čárových kódů nebo QR kódů. Existují i aplikace, které na obraze pořizovaném kamerou v reálném čase, dokáží přeložit nápisy z cizích jazyků.
- Ve fotoaparátech lze pomocí počítačového vidění detekovat obličeje nebo nalézt na fotce oči a odstranit červenou barvu, která vznikne při focení s bleskem.
- V lékařství lze pomocí počítačového vidění analyzovat výsledky různých vyšetření, jako jsou snímky z rentgenů, ultrazvuku nebo počítačové tomografie.
- Ve vozidlech nalezne uplatnění v systémech na detekci chodců, vozidel a dopravního značení. Detekuje dopravní značky nebo sleduje pruh, ve kterém se vozidlo nachází.
- V herních konzolích, jako jsou například Kinect a PlayStation Move, snímají pohyby člověka stojícího před kamerou a jeho pohyby přenášejí na obrazovku.
- Důležitou oblastí, kde se počítačové vidění často využívá, je armáda. Ta ho používá v bezpilotních letadlech či při prozkoumávání neznámých terénů za pomoci robotů.

3 Zpracování obrazu

V této kapitole budou popsány barevné modely, které se využívají při zpracování obrazu. Jsou zde porovnány výhody a nevýhody jednotlivých barevných modelů při detekci dopravního značení. Při psaní této kapitoly bylo čerpáno především z knih [2] a [5]. Dále jsou zde popsány jednotlivé kroky pro obecné zpracování obrazu. Jedná se o:

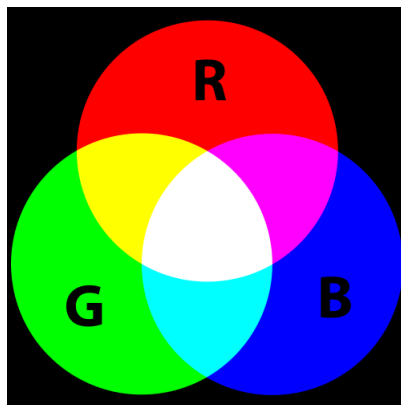
- snímání a digitalizace obrazu,
- předzpracování obrazu,
- segmentace obrazu,
- popis objektu,
- klasifikace.

3.1 Barevné modely

Při zpracování videozáznamů program využívá pro různé operace následující barevné modely, které jsou popsány níže.

3.1.1 RGB

RGB model je jeden z nejpoužívanějších barevných modelů. Velmi často se využívá v monitorech, projektorech a podobných zobrazovacích zařízeních. RGB model je tvořen třemi základními barvami, tj. červená (Red), zelená (Green) a modrá (Blue). Zbylých barev z barevného spektra lze dosáhnout adaptivním mícháním těchto základních barev. Pokud je intenzita u všech barev nastavena na nulu získá se černá barva. Bílá barva se získá, pokud se všem třem složkám nastaví maximální intenzita. Adaptivní míchání barev je ukázáno na obrázku 2.



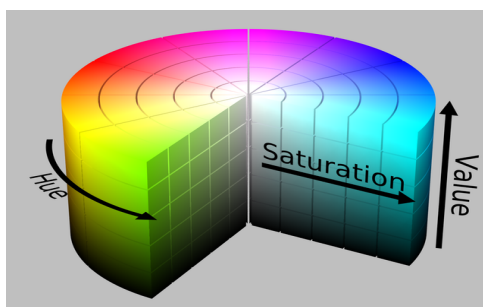
Obrázek 2: Aditivní míchání barev [16]

3.1.2 HSV

Tento barevný model nejvíce odpovídá lidskému vnímání barev. Jedná se o barevný model, kde je barva bodu určena třemi složkami. Nejedná se však o tři základní barvy, jako tomu je v případě modelu RGB. Tyto složky mají zcela jiný význam.

- **Hue** - barevný tón neboli odstín. Tato hodnota udává, která barva v daném bodě převládá. Je udávána ve stupních (0 až 360 stupňů).
- **Saturation** - sytost. Představuje množství šedi vzhledem k odstínu, udává se v % (v rozmezí od 0% pro šedou barvu do 100% pro plně sytou barvu).
- **Value** - jasová hodnota. Udává světlost, nebo tmavost barvy. Vyšší hodnota jasu značí, že barva odráží více světla. Je udávána v %, kde bílá barva má hodnotu jasu 100% a černá barva 0%.

Jak vypadá grafické znázornění modelu HSV je zobrazeno na obrázku 3.



Obrázek 3: Grafické znázornění HSV[12]

Převod RGB modelu na HSV je dán následujícími vzorci:

$$H = \begin{cases} 60^\circ \cdot \left(\frac{G-B}{V-\min(R,G,B)} \right) & V = R \\ 60^\circ \cdot \left(\frac{B-R}{V-\min(R,G,B)} \right) + 120^\circ & V = G \\ 60^\circ \cdot \left(\frac{R-G}{V-\min(R,G,B)} \right) + 240^\circ & V = B \end{cases} \quad (1)$$

$$S = \begin{cases} 0 & V = 0 \\ \frac{V-\min(R,G,B)}{V} & V \neq 0 \end{cases} \quad (2)$$

$$V = \max(R, G, B) \quad (3)$$

Rovnice 1, 2 a 3 popisují převod z barevného modelu RGB na model HSV. Hodnoty R , G , B popisují jednotlivé složky barevného modelu RGB. Jak už je ze vzorců patrné, převod obrazu z barevného modelu RGB na model HSV vyžaduje více operací než převod na obraz ve stupních šedi.

Výhodou barevného modelu HSV je, že umožňuje oddělit informace o barvě od informace o jasové hodnotě daného bodu. Těto výhody se využívá při zkoumání barvy jednotlivých bodů, jelikož není ovlivňovaná jasovou hodnotou.

3.1.3 Obraz ve stupních šedi

Obraz ve stupních šedi, na rozdíl od RGB obrazu, obsahuje pro každý bod pouze jednu jasovou hodnotu. Lidské oko rozlišuje jednotlivé složky RGB s různou citlivostí, proto se nedá využít pouhého sečtení jednotlivých složek, ale pro převod se využívá váhových koeficientů:

$$f = 0.299 R + 0.587 G + 0.114 B, \quad (4)$$

kde f je výsledná úroveň jasu v šedotónovém obraze a R , G , B jsou vynásobeny významem úrovně jasu jednotlivých složek daného bodu v barevném obraze. Srovnání barevného (RGB) obrazu a obrazu ve stupních šedi je ukázáno na obrázku 4.



Obrázek 4: Porovnání barevného obrazu a obrazu v odstínech šedi

3.1.4 Binární obraz

Je to digitální jedno bitový obraz tvořený pouze dvěma odstíny šedé barvy. Na binárním obraze je typicky každý pixel tvořen černou (označující oblast pozadí) nebo bílou barvou (oblast objektu). Proto se mu také říká černobílý obraz. Černobílého obrazu lze dosáhnout pomocí prahování obrazu (thresholding). Jak vypadá takový binární obraz je zobrazeno na obrázku 5.



Obrázek 5: Černobílý obraz

3.1.5 Barevný obraz vs. obraz ve stupních šedi

Při detekci vodorovného dopravního značení je vhodnější použít obraz ve stupních šedi. Vozovka, která je snímána kamerou, je ve většině případů tvořena pouze bílou a šedou barvou. Proto je zbytečné uchovávat při zpracování videa informaci o barvě. Navíc u barevného modelu RGB není jasová hodnota explicitně oddělena od odstínu barvy, a proto je složitější určit intenzitu jasu daného pixelu. Pokud je potřeba pomocí segmentace na základě intenzity jasu získat binární obraz, je lepší využít obraz ve stupních šedi.

3.2 Snímání a digitalizace obrazu

Pro zpracování a rozpoznání obrazu je důležité převést obraz reálného světa do digitální formy. Při snímání dochází k převodu vstupních veličin na elektrický signál spojitý v čase. Příklady vstupních veličin jsou jas, ultrazvuk, tepelné záření atd. Snímání může probíhat v jednom nebo více spektrálních pásmech. Spektrální pásmo je elektromagnetické záření s vymezeným rozsahem vlnové délky. Pro zachycení barevného obrazu stačí tři spektrální složky (červená, zelená, modrá). Tato bakalářská práce se o barevném obraze více zmiňuje v kapitole 3.1.

Při digitalizaci je vstupní spojitý signál převeden do diskrétního tvaru, následně je vstupní signál vzorkován a kvantován. Vzorkování znamená, že vstupní signál je rozdělen na nezbytné množství malých částí, neboli vzorků, které jsou následně zpracovávány. Tímto způsobem zpracování lze získat konečný počet vzorků, které však stále obsahují velké množství informací. Proto je nutné aplikovat další proces tzv. kvantování signálu. Cílem kvantování signálů je zaokrouhlit hodnoty signálu, získaného při vzorkování, na předem definované kvantovací úrovni.

Výsledkem digitalizace je matice přirozených čísel, která reprezentuje obraz. Základní a nedělitelnou jednotkou digitálního obrazu je pixel odpovídající jednomu prvku z této matice.

3.3 Předzpracování obrazu

Základními cíli předzpracování obrazu je potlačení šumu a zkreslení, které vzniklo během digitalizace a přenosu obrazu. Předzpracování obrazu se dále využívá při snaze o zvýraznění určitých rysů v obraze, které jsou pak dále zpracovávány.

Mezi základní a často využívané metody předzpracování obrazu patří:

- filtrace a ostření,
- geometrické transformace,
- jasové transformace,
- matematické morfologie.

3.3.1 Konvoluce a vyhlazování

Při filtraci obrazu, jako je např. vyhlazování obrazu nebo zvýraznění hran, se využívá operace zvaná konvoluce. Při konvoluci se používá tzv. konvoluční maska. Jedná se o matematickou operaci s maticí, která kombinuje dvě vstupní matice, aby získala matici třetí. První matice udává použitý obraz, který je určený k úpravě. Druhá matice je pak tzv. maska, která udává, jaký efekt s obrazem má být proveden. Výsledkem je pak třetí matice.

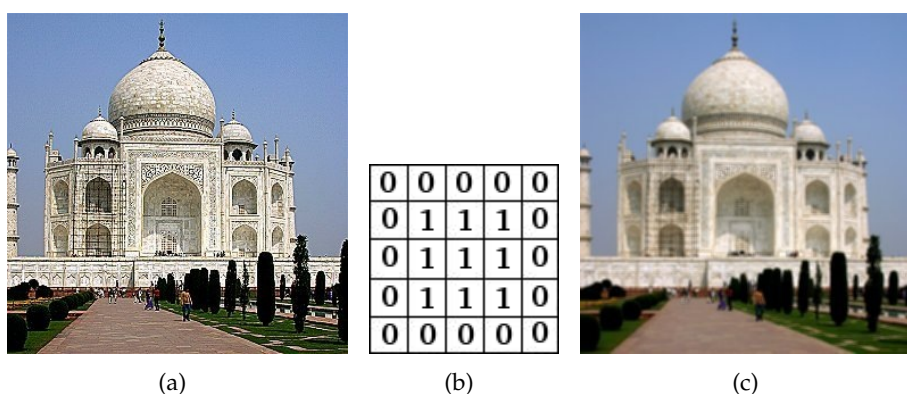
V grafických editorech se nejčastěji používá maska o velikosti 3x3 nebo 5x5. Tyto velikosti jsou dostatečné pro většinu požadovaných efektů.

Všechny pixely obrazu jsou postupně zpracovávány tímto filtrem. Hodnota aktuálního pixelu středu masky je vynásobena hodnotami osmi sousedních pixelů. Výsledné hodnoty jsou poté sečteny a přiřazeny aktuálnímu pixelu.

$$g(x, y) = h \cdot f = \sum_{s=-s_{max}}^{s_{max}} \sum_{t=-t_{max}}^{t_{max}} h(s, t) f(x - s, y - t), \quad (5)$$

kde s_{max} a t_{max} vyjadřují polovinu rozměrů masky, $f(x, y)$ je vstupní obraz a $h(s, t)$ je tzv. jádro. Výsledný obraz je pak označen $g(x, y)$.

Pomocí konvoluce lze obrázek rozmazat, zaostřit, detekovat hrany nebo vytvořit reliéf. Jak vypadá rozmazání obrazu je uvedeno na obrázku 6. Na obrázku 6(a) je zobrazen původní snímek, obrázek 6(b) ukazuje, jak vypadá použité jádro, pomocí kterého se dosáhne efektu rozmazání. Výsledek tohoto efektu je znázorněn na obrázku 6(c).



Obrázek 6: Konvoluce- rozmazání obrazu[11]

3.3.2 Geometrická transformace

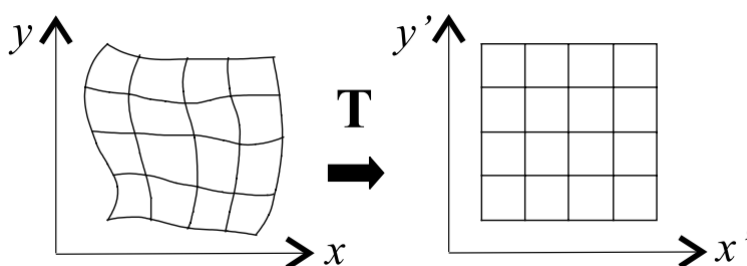
Geometrická transformace T_G obrazu je vektorová funkce, která transformuje bod v rovině (x, y) do bodu (x', y') . Situaci ilustruje obrázek 7. Transformace T_G je definována složkovými vztahy:

$$x' = T_x(x, y), y' = T_y(x, y) \quad (6)$$

V případě použití transformací jako jsou rotace, posunutí nebo změna měřítka jsou transformační vztahy T_x a T_y známy předem. Druhou možností je pak hledat transformační vztah na základě původního a transformovaného obrazu. Při hledání transformační matice se využije několika známých bodů, které si na původním a transformovaném obraze odpovídají. Na základě těchto odpovídajících si bodů se poté vytvoří transformační matice.

Jako příklad geometrické transformace lze uvést družicové snímky Země, kde její typické zakřivení nesouhlasí s běžně užívanými mapami.

Jak vypadá taková transformace je zobrazeno na obrázku 7.



Obrázek 7: Geometrická transformace v rovině[4]

3.4 Segmentace obrazu

Jedním z nejdůležitějších kroků vedoucích k analýze obsahu na obrázku je segmentace obrazu. Jeho snahou je rozčlenit obraz na části, které nějak souvisí s předměty či oblastmi (objekty) reálného světa a pozadí. Segmentaci dělíme na kompletní nebo částečnou. Dá se říci, že u kompletní segmentace se výsledné oblasti shodují s objekty na vstupním obraze. Pokud tomu tak není, a výsledné oblasti se přímo neshodují s objekty na vstupním obraze, pak jde o částečnou segmentaci. Velkou výhodou segmentace obrazu je výrazné snížení objemu zpracovaných dat. Mezi problémy, ovlivňující segmentaci, patří nejednoznačnost obrázkových dat, která je často doprovázena informačním šumem.

Pro segmentaci lze použít některou z těchto metod:

- segmentace prahováním,
- segmentace na základě detekce hran,
- segmentace narůstáním oblastí,
- segmentace srovnáním se vzorem.

3.4.1 Segmentace prahováním

Thresholding neboli prahování je nejstarší a nejjednodušší metoda pro segmentaci obrazu. Zároveň je také, díky nízké výpočetní náročnosti, nejrychlejší segmentační metodou a bývá proto často používána při zpracování obrazu v reálném čase. Pomocí zvolené jasové konstanty neboli prahu lze oddělit objekty od pozadí.

Definice 3.1 *Prahování je transformace vstupního obrazu f na výstupní (segmentovaný) binární obraz g podle vztahu:*

$$g(x, y) = \begin{cases} 1 & \text{pro } f(x, y) \geq T, \\ 0 & \text{pro } f(x, y) < T \end{cases} \quad (7)$$

Pro vzorec 7 platí, že T je předem zvolená konstanta nazvaná práh, $g(x, y) = 1$ pro obrazové elementy náležející po segmentaci objektům a $g(x, y) = 0$ pro elementy pozadí (nebo naopak).

Metoda segmentace prahováním se používá pro převedení obrazu, který obsahuje více úrovní jasu (odstíny šedi) na obraz, ve kterém se vyskytují dvě jasové úrovně (černá a bílá). Pokud je hodnota jasu v daném bodě větší než práh (threshold), je danému bodu přiřazena hodnota 1 (případně 255). Prahování může být globální (statické) nebo lokální (adaptivní).

U globálního prahování je zvolená hodnota prahu, která platí pro celý obraz. Vhodné použití tohoto typu je u obrazů, kde je neměnný jas, jako jsou např. krevní buňky, psaný text apod. Globální práh je určován z celého obrazu:

$$T = T(f) \quad (8)$$

Jak vypadá segmentace obrazu prahováním je zobrazeno na obrázku 8. Na obrázku 8(a) je znázorněn původní obrázek a na obrázku 8(b) pak výsledek po prahování.

Na výsledném obrázku je vidět, že obsahuje pouze dvě barvy a to bílou a černou. Bílá barva byla přiřazena oblastem, které měly hodnotu jasu vyšší, než byl zvolený práh, a černou barvou pak byly vykresleny zbylé oblasti.



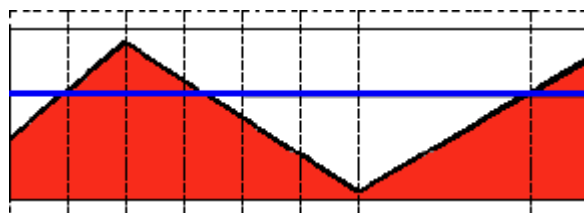
Obrázek 8: Segmentace- prahováním[17]

Existuje pět typů prahových operací:

- Threshold Binary - pixelům s hodnotou větší, než je hodnota prahu, bude přidělena hodnota 1. Pixelům s hodnotou menší pak bude přiřazena 0.
- Threshold Binary, Inverted - tato metoda je inverzní metodou k metodě Threshold Binary.
- Truncate - ořízne hodnoty pixelu, které jsou větší než hodnota zvoleného prahu, na hodnotu prahu. Ostatní hodnoty pixelu zůstanou nezměněny.
- Threshold to Zero - pixelům s hodnotou větší než hodnota prahu ponechá původní hodnotu. Ostatní pixelům nastaví hodnotu na 0.
- Threshold to Zero, Inverted - jedná se o inverzní metodu k metodě Threshold to Zero.

Na obrázku 9 je ukázána intenzita jednotlivých pixelů. Modrá čára reprezentuje zvolenou hodnotu prahu. Jak vypadá prahování vzorového obrázku, je popsáno níže.

Pro následující vzorec 9 platí, že T je předem zvolená hodnota prahu, $f(x, y)$ vyjadřuje intenzita pixelu vstupního obrazu, $g(x, y)$ vyjadřuje intenzitu pixelu výstupního obrazu a $maxVal$ je nenulová hodnota, která bude přiřazena bodu, pokud je splněna podmínka (nejčastěji to bývá 255).



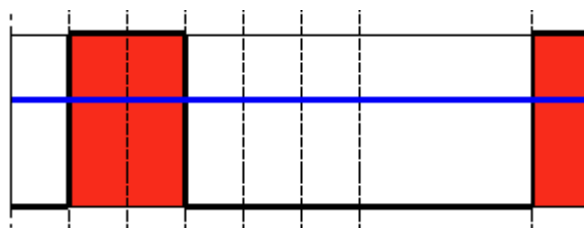
Obrázek 9: Thresholding[15]

Threshold Binary

Tato prahovací metoda může být vyjádřena ze vzorce 7.

$$g(x, y) = \begin{cases} maxVal & \text{pro } f(x, y) > T \\ 0 & \text{jinak} \end{cases} \quad (9)$$

Ze vzorce 9 vyplývá: jestliže je intenzita pixelu $f(x, y)$ větší než hodnota prahu, pak je intenzita nového pixelu $g(x, y)$ nastavena na $MaxVal$. Pokud je tomu jinak nastaví se na 0.



Obrázek 10: Threshold Binary[15]

Na obrázku 10 je vidět, že po použití prahovací metody Threshold Binary vznikne binární obraz. Pixely s intenzitou větší než je zvolená hodnota prahu, budou mít hodnotu intenzity $maxVal$ a pixely, které měly intenzitu menší, budou mít hodnotu 0.

3.4.2 Adaptivní prahování

Při lokálním prahování je vstupní obraz rozdělen na jednotlivé části, které se vzájemně nepřekrývají. U adaptivního prahování se hodnota prahu v každé části obrazu vypočte zvlášť, čímž lze dosáhnout lepšího výsledku. Výhodou lokálního prahování je, že dokáže vykompenzovat nerovnoměrné nasvícení obrazu. Adaptivní metoda prahování umožňuje použití pouze dvou typů prahových operací a to Threshold Binary a Threshold Binary Inverted. Lokální práh je určen pomocí funkcí polohy:

$$T = T(f, f_c), \quad (10)$$

kde f je celý obraz f_c je část obrazu, pro niž je práh určován.

Metody pro výpočet hodnoty prahu:

Adaptive thresh mean C

Všechny pixely v této oblasti mají stejnou váhu.

$$T(x, y) = blockSize \cdot blockSize - C, \quad (11)$$

kde $T(x, y)$ je hodnota prahu, $blockSize^2$ je oblast okolí bodu a C je konstanta.

Adaptive thresh Gaussian C

Pixely v regionu (x, y) jsou váženy podle Gaussovy funkce, tj. podle jejich vzdálenosti od tohoto středového bodu.

3.5 Popis objektu

Po segmentaci objektu přichází na řadu popis nalezených objektů v obraze. Toto lze provést buď kvantitativně (pomocí číselných charakteristik) nebo kvalitativně (pomocí relací mezi objekty). Jak bude objekt popsán závisí na skutečnosti, na co se popis bude používat. Za nejčastěji používaný způsob popisu lze považovat popis na základě velikosti (plochy) objektu, tj. počtu jemu odpovídajících pixelů v obraze.

3.6 Klasifikace

Posledním krokem pro zpracování obrazu je klasifikace objektů. Jedná se o zařazení objektů nalezených v obraze do skupiny předem známých tříd (např. kulaté a hranaté). Metody klasifikace jsou rozděleny do dvou základních skupin a to příznakové a strukturální rozpoznávání.

Příznakové metody rozpoznávání fungují na principu využití příznaků objektů. Těchto příznaků by mělo být co nejméně, ale měly by objekty co nejlépe popisovat. Využitím více příznaků vznikne tzv. vektor příznaků. Tento vektor pak nese veškeré podstatné informace o daném objektu. Při následném rozpoznávání se pak využívá pouze těchto vektorů. Za příklad příznaků můžou být považovány pravoúhlost nebo kulatost.

Strukturální metody rozpoznávání fungují na principu, kdy obraz je složen ze základních popisných elementů tzv. primitiv, jejich vlastností a vztahů mezi nimi. V praxi se využívá jak příznakové tak strukturální metody současně, jelikož se vzájemně doplňují.

3.6.1 Klasifikační metody

Pro klasifikaci objektu lze použít několik metod. Mezi často používané příznakové metody rozpoznávání patří Neuronové sítě, které se snaží napodobit fungování lidské nervové soustavy. Neuronové sítě jsou schopné se učit na základě zpětné vazby. Dalšími příklady klasifikačních metod jsou K-means, která třídí data do shluků na základě jejich vlastností. SVM je metoda, která při klasifikaci hledá nadrovinu, pomocí které se v oblasti s příznaky optimálně rozdělí trénovací data. Metoda SURF se skládá ze dvou částí. V první části se v obraze naleznou klíčové body (např. rohy). Druhým krokem je pak výpočet deskriptoru z okolí klíčového bodu, pomocí kterého se získají příznakové vektory. Tyto vektory jsou pak použity při porovnávání dvou obrázků. Jednou z možností jak klasifikovat objekty, je použití metody MatchTemplate, která je založena na podobnosti dvou obrázků.

MatchTemplate

Metoda MatchTemplate funguje na principu porovnávání dvou obrázků. Jednou z možností využití této metody je i hledání obrázků jako součást toho druhého. Podmínkou je, aby hledaný obraz neměl větší rozměry než obraz, na kterém se hledá.

Tato metoda má čtyři parametry:

- vstupní obraz, ve kterém se bude vyhledávat,
- šablonu objektu neboli obraz, který má být vyhledán,
- výstupní pole, do kterého se zapisuje pravděpodobnost výskytu,
- posledním parametrem je pak metoda, která se využije pro vyhledávání, posouvání šablony po obraze a výpočet podobnosti.

Pro výpočet podobnosti existují tři základní metody:

- Square difference matching method
 - suma čtverců rozdílu mezi hodnotou pixelu v obraze a šabloně,

$$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2, \quad (12)$$

kde R značí výstupní parametr, tedy pole pravděpodobnosti výskytu, T značí obrázek šablony a I značí obrázek, na kterém se objekt hledá.

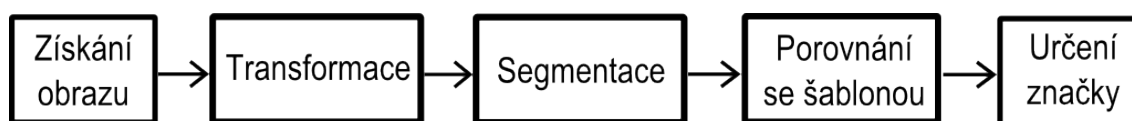
- Correlation matching method
 - suma čtverců korelace pixelů v obraze a šabloně.
- Correlation coeff. matching method
 - suma čtverců korelačních koeficientů pixelů v obraze a šabloně.

Uvedený vzorec je pouze pro metodu Square difference matching method, protože byla použita v aplikaci na detekci vodorovného dopravního značení. Zbylé dvě metody jsou podrobněji popsány v knize [1] nebo na internetových stránkách [14], kde se nachází dokumentace k OpenCV.

Všechny tři metody mají i své normalizované varianty. Ty zajistí, že hodnota podobnosti bude převedena na interval $< 0, 1 >$. Této normalizace se využívá, pokud je potřeba použít filtr pro zajištění alespoň určitého procenta shody. U první metody, tedy Square difference matching method, má nejlepší shoda minimální hodnotu (u normalizované verze je to hodnota 0). U zbylých dvou metod je tomu naopak a nejlepší shoda má maximální hodnotu (u normalizované verze je to hodnota 1).

4 Vlastní implementace řešení

Čtvrtá kapitola je zaměřena na vlastní implementaci řešení. Konkrétně zde bude popsáno, jak probíhá získání a přehrávání videa a předzpracování jednotlivých snímků z videosekvence. Pro získání oblasti výskytu značek byla použita segmentace obrazu. Tato kapitola rovněž bude popisovat možnosti, jak potlačit šum v obraze a následné porovnání snímku se šablonou dopravních značek, které je program schopen rozpoznat. Jak tyto kroky za sebou následují, zobrazuje obrázek 11.



Obrázek 11: Jednotlivé kroky zpracování obrazu

OpenCV

OpenCV je multiplatformní knihovna programovacích funkcí pro zpracování obrazu v reálném čase. Je zdarma uvolněná pro akademické i komerční použití. Používá se například pro kontroly v hornictví, interaktivní umění nebo u robotů. Aplikace optimalizované v C/C++ mohou využívat vícejádrové procesory.

4.1 Získání obrazu

Snímky z videosekvence vstupují do algoritmu jednotlivě. Pro zrychlení procesu zpracování obrazu, jsou všechny snímky videa při vstupu do programu zmenšeny na rozlišení 320x240 pixelů. Toto rozlišení bylo zvolené na základě testování programu. Při rozlišení 320x240 došlo ke zmenšení oblasti pozadí, které se vyskytovaly uvnitř oblasti zájmů, a proto byl program schopen při porovnávání lépe určit, o kterou dopravní značku se jedná. Při vyšším rozlišení by se čas zpracování zvýšil, čímž by nebylo dosaženo detekce v reálném čase.

4.2 Transformace obrazu

Při použití metody MatchTemplate bylo důležité, aby byly oba obrazy, tj. obraz se šablonami a vstupní snímek z videa, zobrazeny pod stejným úhlem. Jelikož kamera, umístěná ve vozidle, sleduje vozovku pod úhlem cca 45 stupňů, je potřeba transformovat obraz na úhel 90 stupňů.

Prvním krokem při transformaci je získání transformační matice. Tu je možno získat pomocí metody `GetPerspectiveTransform`. Tato metoda má tři parametry:

- prvním parametrem jsou souřadnice vrcholů čtyřúhelníku ve zdrojovém obrázku, tj. oblast kterou chceme transformovat,
- druhým parametrem jsou souřadnice odpovídajících vrcholů čtyřúhelníku v cílovém, tj. transformovaném obraze,
- posledním a zároveň výstupním parametrem je pak matice se třemi sloupci a třemi řádky, do které se uloží výsledná transformační matice potřebná k transformaci zdrojového čtyřúhelníku na transformovaný čtyřúhelník.

Aby se zdrojový obraz snímající silnici změnil z pohledu z auta na pohled z ptačí perspektivy, musí se obraz na výšku roztáhnout a zároveň se musí spodní hrana obrazu zmenšit. V případě aplikace na detekci vodorovného dopravního značení byla vybraná oblast, která se má transformovat. Tato oblast byla vybraná tak, aby se na ní zbytečně nevyskytovalo nebe a zároveň se zmenšila oblast, kterou na videu zabírala kapota vozidla. Jak taková transformace vypadá je zobrazeno na obrázku 12, kde zdrojový obraz je označen písmenem (a) a výsledný, tj. transformovaný písmenem (b).



Obrázek 12: Transformace obrazu

4.3 Odstranění šumu

Vyhlazování, někdy se také používá i termín rozmazávání, je jednoduchou a často využívanou operací pro odstranění nežádoucího šumu v obraze. Jednou z častou používaných metod pro vyhlazování je metoda Gaussovo vyhlazování. Kromě ní se často používá i metoda vyhlazování průměrováním.

Metoda Gaussovo vyhlazování funguje na principu, že koeficienty blíže středu masky mají vyšší váhu a odpovídají proto hodnotám na Gaussově křivce. Masky je dána dvěma vstupními parametry. Tyto parametry musí být kladné liché číslo. Masky určuje oblast, ze které se bude vypočítávat jas pixelu, který leží uprostřed této oblasti.

4.4 Segmentace dopravních značek

Při detekci vodorovného dopravního značení v obraze lze vycházet z faktu, že dopravní značení je na silnicích vyznačené bílou barvou nebo, v případě přechodného dopravního značení, žlutou barvou. Díky tomuto faktu lze v obraze hledat pouze oblasti této barvy. Pro detekci těchto oblastí lze využít jednu z metod na prahování obrazu, které jsou podrobněji popsány v kapitole 3.4.1. V aplikaci na detekci vodorovného dopravního značení bylo potřeba převést barevný obraz na obraz ve stupních šedi. Důvodem této konverze byly požadavky metody na adaptivní prahování, která jako vstupní parametr požaduje monochromatický obraz. Této přeměny lze dosáhnout pomocí metody na konvertování barev, díky které se převede barevný, třísložkový obraz (red, green, blue), na obraz monochromatický. Ten má pouze jednu složku a tou je jas. Na tento obraz se pak může aplikovat prahování.

Při zpracování lze použít globální prahování nebo lokální (adaptivní) prahování. Při psaní programu bylo nejprve použito globální prahování, pomocí kterého se dosáhlo segmentace oblasti zájmu. Jelikož tyto oblasti byly často různě deformovány a zároveň docházelo na videu k častým změnám jasu, bylo tedy vhodnější v tomto případě použít adaptivní prahování, pomocí kterého bylo dosaženo lepších výsledků.

Metoda adaptivního prahování má několik parametrů:

- vstupní obrázek,
- výstupní obrázek,
- maxValue - nenulová hodnota přiřazená pixelu, pokud je splněná podmínka prahovací operace,
- hodnota prahu, která se vypočítá pomocí metod na výpočet hodnoty prahu (viz kapitola 3.6.1),
- typ prahovací operace (Threshold Binary a Threshold Binary, Inverted),
- velikost bloku - okolí bodu, které je použito pro výpočet hodnoty prahu,
- konstanta, která je použita u metod pro výpočet prahu.

Na obrázku 13 je zobrazeno, jak vypadá segmentace obrazu, který vznikl po předchozí transformaci (kapitola 4.2).



Obrázek 13: Segmentace obrazu prahováním

Z obrázku 13 je patrné, že oblasti, které měly hodnotu jasu vyšší než byla nastavená hodnota prahu, mají bílou barvu. Oblasti, které vznikly na okrajích obrázku 13(a), budou v dalších krocích programu ořezány.

Pro odstranění malých oblastí pozadí na černobílém obrázku, které jsou uvnitř oblastí zájmu, se použije dilatace. Dilatace znamená, že každý pixel oblasti zájmu se rozšíří o jeden pixel vedle a tím odstraní mezery, které mají tloušťku jednoho pixelu.

Pro odstranění větších mezer v oblasti můžeme použít dilataci opakovaně.

Pro zmenšení objektů, na které byla použita dilatace, na původní rozměry objektu lze použít erozi. Při erozi se pixely ubírají z obrysu oblasti zájmu. Pokud je oblast příliš úzká, zmizí z obrazu úplně. Erozi lze rovněž použít opakovaně.

Jak vypadá redukce takových oblastí je zobrazeno na obrázku 14. Na obrázku 14(a) lze vidět oblasti pozadí uvnitř šipky, který je potřeba odstranit. Obrázek 14(b) pak ukazuje výsledný obraz po odstranění těchto oblastí.



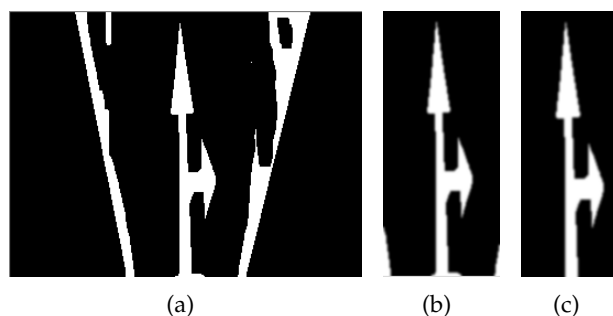
Obrázek 14: Použití dilatace a eroze

Na obrázek 14(a) byla nejprve použita dilatace, aby se odstranily oblasti o velikosti jednoho pixelu. Druhým krokem pak bylo použití eroze, která vrátila oblasti dopravní značky původní velikost.

4.5 Oblast zájmu a zmenšení obrazu

Při zpracování obrazu je vhodné zmenšit rozměry obrazu tím, že se odřízne z obrazu oblast, ve které se nepočítá s výskytem oblasti zájmu. V případě obrazu (zpracováván aplikací na detekci vodorovného dopravního značení) tak jde o oblasti, které se vytvořily při předzpracování obrazu a jeho transformací. Kvůli změně úhlu pohledu byla zúžená dolní hrana obrazu na jednu třetinu původní šířky. Hledaná oblast zájmu se tedy vyskytuje v prostřední třetině obrazu. Tato část z obrazu se vyřízne, a následně v dalších krocích se pracuje už jen s ní. Rozměry této oblasti byly určeny na základě experimentu při testování na několika videozáznamech.

Na obrázku 15 je zobrazeno, jak probíhá výběr oblasti zájmu.



Obrázek 15: Vybrané oblasti zájmu

Nejprve byla z obrázku 15(a), který byl rozdělen na třetiny, vybraná prostřední část. Tato část je zobrazena na obrázku 15(b). Druhé oříznutí obrazu proběhlo na základě souřadnic, které byly získány experimentálním testováním tak, aby na obrázku zůstala pouze požadovaná oblast. Výsledek tohoto oříznutí je zobrazen na obrázku 15(c).

Důležitým krokem pro porovnání dvou obrázků nebo nalezení jednoho, jako součást toho druhého, je proces normalizace. Pod pojmem normalizace se rozumí, že obrázek, který je porovnáván se šablonou, musí mít stejné rozměry jako jednotlivé obrázky dopravních značek na šabloně. To je vynuceno tím, že metoda, která se používá pro porovnání dvou obrázků, nepovoluje, aby hledaný obrázek byl větší než obrázek, na kterém ho hledáme. To také znamená, že je nutné zajistit aby objekt, který se nachází na hledaném obrázku a jemu odpovídající objekt na šabloně měl stejnou velikost.

4.6 Vytvoření šablon dopravních značek

Pro nalezení a rozpoznání vodorovných dopravních značek ve videu bylo potřeba vytvořit šablonu, která bude obsahovat všechny značky, které je schopen program rozpoznat a správně určit. Všechny šipky, které jsou ukázány na obrázku 1, byly zmenšeny na stejnou velikost. Tato velikost byla stanovená na rozlišení 100 x 100 pixelů. Zároveň byly šipky upraveny tak, aby na zmenšeném obrázku zabíraly odpovídající velikost. Do šablony byly také přidány některé další dopravní značky, jako jsou například nápis POZOR. Ten se objevuje na silnici v místech, kde vozovka křížuje tramvaj. Dále byly přidány i značky „Pozor, přechod pro chodce“ a dopravní značka znázorňující samotný přechod. Tyto značky byly do šablony přidány z důvodu snížení chybovosti správného určení dopravní značky. Všechny tyto značky byly následně

pomocí grafického editoru spojeny do jednoho velkého obrázku. Tento obrázek má na výšku rozměr 100 pixelů a na šířku součet šířek všech značek. Jak tato šablona vypadá je ukázáno na obrázku 16. Jak je z obrázku patrné, některé dopravní značky, jednalo se především o šipku rovně a předběžnou šipku, byly získány z testovacího videa, aby lépe odpovídaly tvaru šipky.



Obrázek 16: Šablona dopravních značek

Šablona je při rozpoznávání rozdělená do 8 částí:

- první část šablony tvoří černý čtverec, který znázorňuje prázdnou vozovku,
- druhá část je šipka doprava,
- třetí část je šipka rovně a doleva,
- čtvrtá část je šipka rovně a doprava,
- pátá část je šipka rovně,
- šestá část je šipka značící spojování dvou pruhů, ta je na šabloně umístěná dvakrát.
- sedmá část značí šipku doleva,
- poslední část šablony tvoří dopravní značky pro přechod pro chodce a nápis POZOR. Jsou to ty značky, které nechceme detekovat.

4.7 Porovnání objektu na obrázku se šablonami

Porovnávání snímku z videosekvence se šablonou se provádí pomocí metody `matchTemplate`. Jak už bylo uvedeno v kapitole 3.6.1, tato metoda má čtyři parametry:

- prvním parametrem je šablona dopravních značek,
- druhým parametrem je snímek pořízený z videozáznamu,
- třetím parametrem je pole pravděpodobnosti výskytu,
- poslední parametr je nastaven na normalizovanou verzi metody `Square difference Matching`. Tato metoda byla zvolená na základě experimentů na testovaných videozáznamech.

Dalším krokem je pak nalezení minima (maxima) ve výstupní matici. Toto minimum (maximum) lze získat pomocí metody `cvminMaxLoc`. Tato metoda zjistí pozici minimální a maximální hodnoty ve výstupní matici, která se získá pomocí výše uvedené metody `matchTemplate`. Která hodnota se použije, záleží na zvolené metodě pro výpočet shody. V případě této aplikace se tedy pracuje s minimální hodnotou, a proto se využije pozice minimální hodnoty. Na této pozici se následně vykreslí čtverec o rozměrech 100 x 100 pixelů. Tato velikost odpovídá rozlišení jednotlivých značek. Pozice tohoto čtverce se pak využívá při určení, o kterou dopravní značku se jedná. Jak už bylo popsáno v kapitole 3.6.1, pro normalizované varianty je vhodné si zvolit určitý práh. Pomocí tohoto prahu se zajistí minimální procento podobnosti. V přiloženém programu je tento práh nastaven na hodnotu 0.46, což znamená, že podobnost musí být alespoň z 54%. Hodnota 0.46 byla zvolena na základě testování tak, aby aplikace rozpoznala a správně určila co největší množství dopravních značek přesně. Pokud by byla hodnota větší, docházelo by k občasnému špatnému určení dopravní značky. Rovněž by aplikace mohla jako dopravní značku považovat i oblasti, ve kterých se dopravní značka vůbec nenachází (např. přerušovaná čára by mohla být určena jako šipka rovně).

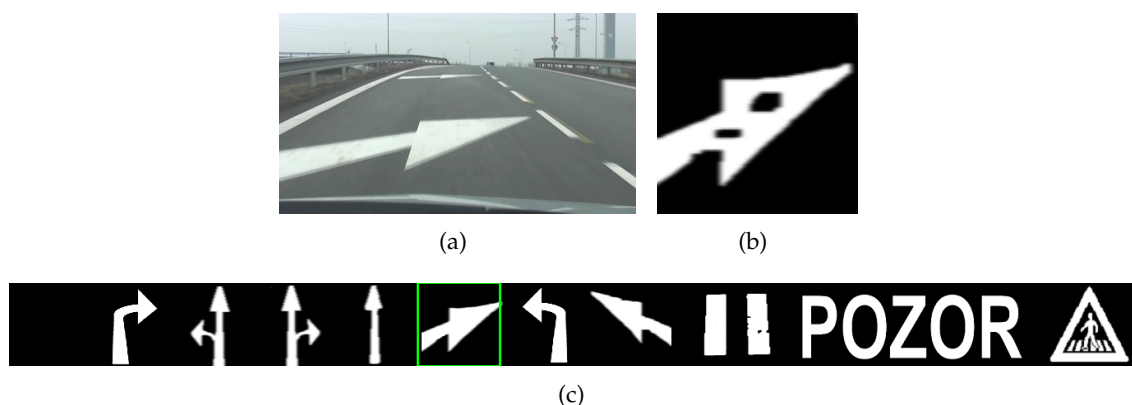
4.8 Určení dané dopravní značky

Program rozpozná a dokáže určit 6 dopravních značek, které to jsou je ukázáno na obrázku 1.

Pro určení dopravní značky bylo vytvořené pole. Toto pole má alokovanou velikost 8. Vleze se do něj tedy 8 hodnot, kde jeden záznam v poli odpovídá jedné části ze šablony. Do tohoto pole budou následně zapisovány nejlepší shody s každou značkou. Pomocí pozice čtverce, který je popsán výše, se zjistí, ve které části šablony se nachází nejlepší shoda. Hodnota této shody se pak porovná s hodnotou v poli, která odpovídá dané části šablony. Defaultně je hodnota v poli nastavena na 1 jako minimální shoda. Pokud je aktuální hodnota menší (je tedy dosaženo větší podobnosti) než současná hodnota v poli, tak dojde k zápisu do daného pole a při dalším výskytu podobnosti se pak porovnává hodnota s touto zapsanou hodnotou.

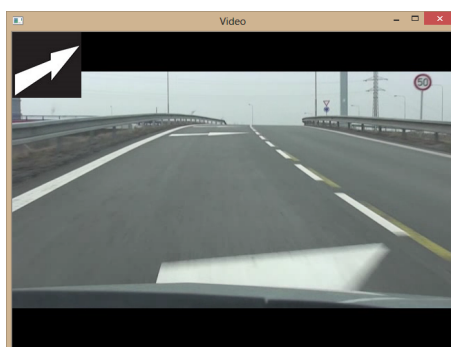
Dopravní značka se začne určovat v momentě, kdy je detekována první část šablony. Tedy ta, která značí vozovku bez dopravní značky. Důvodem je skutečnost, že několik snímku za sebou může obsahovat tutéž (nebo jinou) dopravní značku a je potřeba určit jen tu, která měla nejlepší shodu. Toho je dosaženo pomocí jednoduchého nalezení minima v poli. Jakmile je zjištěna minimální hodnota v poli, je pomocí pozice této minimální hodnoty určeno, o kterou vodorovnou dopravní značku se jedná. Určená

dopravní značka se následně vykreslí do okna s přehrávaným videem. Tato šipka je v okně vykreslená jen po předem stanovenou dobu. Jakmile je dopravní značka určena, dojde k vynulování pole. Tím se opět nastaví minimální hodnota podobnosti, tj. defaultní hodnota, která je stanovená na 1, pro všechny záznamy v poli. Tento proces se opakuje při každé detekci nějaké dopravní značky.



Obrázek 17: Porovnání dopravní značky se šablonou

Na obrázku 17 je ukázáno, jak vypadá snímek, když se porovnává se šablonou. Obrázek 17(a) ukazuje, jak vypadá pohled z automobilu. Na obrázku 17(b) je ukázáno, jak vypadá pohled z automobilu po transformaci, následné segmentaci a zmenšení na velikost 100 x 100 pixelů. Tato velikost je nutná pro porovnání se šablonou. Na obrázku 17(c) je znázorněn zelený čtverec, který už byl v této bakalářské práci popsán. Tento čtverec zvýrazňuje oblast na šabloně, které se obrázek 17(b) nejvíce podobá. Okno aplikace po správné detekci a následném určení dopravní značky je zobrazeno na obrázku 18.



Obrázek 18: Výsledek po detekci a určení dopravní značky

5 Zhodnocení dosažených výsledků

V této kapitole je uvedeno, jak byly jednotlivé operace časově náročné. V tabulce číslo 1 jsou uvedeny dosažené výsledky a kolik procent dopravních značek bylo správně detekováno. Která značka byla detekována správně v nejvíce případech, je uvedeno v tabulce číslo 2.

Testování bylo prováděno na videích pořízených kamerou Panasonic HDC- SD10 z jedoucího automobilu v Ostravě, městské části Výškovice a Mariánské Hory. Auto se po vozovce pohybovalo průměrnou rychlostí 40 km/h. Natáčení probíhalo za denního světla.

Program byl testován na videozáznamech, jejichž celková délka činila 15 minut a 4 sekundy. Na těchto videozáznamech bylo celkem 162 dopravních značek, které je program schopen rozpoznat.

Počet značek	Správně určených	Neurčených	Chybně určených	Správně určených v %
162	146	12	4	90%

Tabulka 1: Vyhodnocení počtu správných určení

Z tabulky číslo 1 je patrné, že program dokázal určit dopravní značku správně v 90% případů. Takto vysoké číslo správného určení dopravní značky může být považováno za vynikající výsledek. Dopravní značky, které nebyly určeny, program ve většině případů poznal, ale nevyhodnotil dostatečnou shodu se šablonou.

U dopravních značek, které byly určeny špatně, se jednalo o to, že:

- Značka nebyla dostatečně kvalitní. Například šipka rovně a doprava se více podobala šipce rovně, protože část znázorňující odbočení doprava byla málo výrazná.
- Část šipky byla mimo detekovanou oblast zájmu.

Jak vypadá taková chybná detekce, kterou program nedokázal správně určit, je zobrazeno na obrázku 19, kde obrázek 19(a) je snímek pořízený z vozidla a na obrázku 19(b) je zobrazeno, jak vypadá daný snímek, když se porovnává se šablonou. Jak je patrné, obrázek 19(b) je více podobný šipce rovně než šipce rovně a doprava.



Obrázek 19: Ukázka chybné detekce

V tabulce číslo 2 je vypsán počet výskytů jednotlivých dopravních značek a u každé značky je uvedeno v kolika procentech případů byla určena správně.

Název	Počet značek	Správně určených	Neurčených	Chybně určených	Správně určených v %
Doleva	9	8	1	0	89%
Rovně a doleva	28	24	4	0	85%
Rovně	49	44	3	2	90%
Rovně a doprava	66	60	4	2	91%
Doprava	3	3	0	0	100%
Spojení	7	7	0	0	100%

Tabulka 2: Výskyt a detekce jednotlivých šípek

Jak už z tabulky 2 vyplývá, nejčastěji se na videozáznamech vyskytovaly dopravní značky: šipka rovně, šipka rovně a doprava, šipka rovně a doleva.

Průměrná časová náročnost jednotlivých operací bude uvedena v následujících bodech:

- získání obrazu - 1 ms,
- předzpracování obrazu (Transformace) - 2 ms,
- segmentace (Detekování bílých objektů na snímku) - 3 ms,
- rozpoznání a určení dopravní značky - 24 ms.

Z uvedených časů je vidět, že většinu času při zpracování obrazu zabírá samotné porovnávání zpracovávaného snímku se šablonou a určení, o kterou dopravní značku se jedná. Celkový čas zpracování jednoho snímku se pak pohyboval kolem 31 ms.

Testování rychlosti zpracování probíhalo na dvou počítačích, které mají rozdílný výkon. Konfigurace obou počítačů je uvedena v tabulce 3.

	Silnější konfigurace	Slabší konfigurace
Procesor	Intel Core i5-3570 (3.80 GHz)	Intel Core i3-330M (2.13 GHz)
Paměť RAM	8 GB	4 GB
Grafická karta	AMD Radeon HD 7850	ATI Mobility Radeon HD5470
Operační systém	MS Windows 8	MS Windows 8

Tabulka 3: Konfigurace počítačů, na kterých probíhalo testování

Hodnoty uvedené výše, byly dosaženy na výkonnějším počítači. Na slabším počítači se celkový čas zpracování pohyboval kolem 65 ms.

Při testování byly použity dvě verze OpenCV knihoven. Jednalo se o verze OpenCV 2.1 a OpenCV 2.2. Na testovaných verzích se časy lišily (verze 2.1 dosáhla lepších výsledků). Pokud se bude vycházet z předpokladu, že zpracovávané video má snímkovou frekvenci 25 FPS (standart v České republice) je tedy potřeba, aby zpracování jednoho snímku (tzn., že doba vykonání veškerých operací, které budou prováděny nad jednotlivými snímky z videosekvence) trvalo maximálně 40 ms. Při delším zpracování začne být obraz zpomalený nebo se začne dokonce trhat.

6 Závěr

Pro svou bakalářskou práci jsem si zvolil téma Detekce vodorovného dopravního značení v obrazech. Práci jsem rozdělil do dvou částí.

V teoretické části, která má dvě kapitoly, jsem se zaměřil v první kapitole na popis obecných informací o dopravním značení a počítačovém vidění. Druhá kapitola je zaměřená na barevné modely, které jsem v programu použil a popisuje jednotlivé kroky, jak probíhá zpracování obrazu.

Druhá část mé bakalářské práce je již zaměřena na vlastní implementaci řešení. Jako první co jsem udělal, bylo získání videí s vodorovným dopravním značením. Tato videa jsem si natáčel sám v Ostravě a to z jedoucího automobilu. Následně jsem si z internetových stránek Ministerstva dopravy stáhnul obrázky vodorovných dopravních značek, které bude program schopen rozpoznat. Pro lepší určení dopravních značek jsem transformoval vstupní videozáznam na pohled „z ptačí perspektivy“. Takto upravený obraz jsem použil pro segmentaci a tím jsem získal požadované dopravní značky. Ty jsem porovnal se šablonou. Výsledky tohoto porovnání jsou uvedeny ve zhodnocení dosažených výsledků mé práce.

Program jsem testoval na 15 minutách videozáznamu. Program dokázal určit dopravní značku správně ve 146 případech z celkového počtu 162 dopravních značek. Dosáhl jsem tím tedy úspěšnosti správného určení 90%.

Rychlost zpracování videozáznamu programem závisí na použité verzi OpenCV knihovny a na výkonu počítače, na kterém je tento program spouštěn.

Jan Štěpán

7 Reference

- [1] BRADSKI, Gary R. *Learning OpenCV*. Sebastopol: O'Reilly, 2008, 555 s. ISBN 978-0-596-51613-0.
- [2] DOBEŠ, Michal. *Zpracování obrazu a algoritmy v C#*. 1. vyd. Praha: BEN - technická literatura, 2008, 143 s. ISBN 978-80-7300-233-6.
- [3] FÍRT, Jaroslav a Radek HOLOTA. *Digitalizace a zpracování obrazu*, 5 s. Dostupné z: <http://home.zcu.cz/~holota5/publ/DigZprO.pdf>.
- [4] HLAVÁČ, Václav. *Jasové a geometrické transformace*. [online prezentace]. Praha: České Vysoké Učení Technické v Praze, 2009,[cit. 2013-04-11]. Dostupný z: <http://cmp.felk.cvut.cz/~hlavac/TeachPresCz/11DigZprObr/18BrightGeomTxCz>.
- [5] HLAVÁČ, Václav a Milan ŠONKA. *Počítačové vidění*. 1. vyd. Praha: Grada, 1992, 272 s. ISBN 80-854-2467-3.
- [6] KHEYROLLAHI, Alireza a Toby P. BRECKON. Automatic real-time road marking recognition using a feature driven approach. *Machine Vision and Applications*, 2012, 1-11. [online]. 2012, roč. 23, č. 1, s. 123-133 [cit. 2013-03-26]. ISSN 0932-8092. DOI: 10.1007/s00138-010-0289-5. Dostupné z: <http://link.springer.com/10.1007/s00138-010-0289-5>.
- [7] SHAPIRO LINDA, G. *Computer Vision*. Vyd. 1. New Jersey: Prentice-Hall, 2001, 580 s. ISBN 01-303-0796-3.
- [8] SOCHOR, Jakub. *Rychlá detekce dopravních značek v obraze*. Brno, 2012. 51 s. Bakalářská práce. Vysoké Učení Technické v Brně. Vedoucí práce Doc. Ing. Adam Herout, Ph.D.
- [9] SOJKA, Eduard. *Digitální zpracování a analýza obrazů*. 1. vyd. Ostrava: VŠB - Technická univerzita, 2000, 133 s. ISBN 80-707-8746-5.
- [10] *Dopravní značení Pardubice: Vodorovné dopravní značky*. [online]. [cit. 2013-04-30]. Dostupné z: <http://www.dzpce.cz/Vodorovne-dopravni-znacky.html>
- [11] *Gimp: Documentation* [online]. [cit. 2013-04-21]. Dostupné z: <http://docs.gimp.org/2.2/cs/plugin-convmatrix.html>.

-
- [12] HSL and HSV: Basic principle. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2013-04-29]. Dostupné z: http://en.wikipedia.org/wiki/HSL_and_HSV
- [13] *Ministerstvo dopravy* [online]. 2006 [cit. 2013-05-01]. Dostupné z: <http://www.mdcr.cz/cs/default.htm>
- [14] *OpenCV: Documentation* [online]. 2010, 2013 [cit. 2013-03-26]. Dostupné z: <http://docs.opencv.org/index.html>.
- [15] *OpenCV: Documentation: Basic Thresholding Operations*. [online]. [cit. 2013-04-29]. Dostupné z: <http://docs.opencv.org/doc/tutorials/imgproc/threshold/threshold.html>
- [16] RGB color model: Additive primary colors. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2013-04-29]. Dostupné z: http://en.wikipedia.org/wiki/RGB_color_model
- [17] Thresholding. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2013-04-29]. Dostupné z: [http://en.wikipedia.org/wiki/Thresholding_\(image_processing\)](http://en.wikipedia.org/wiki/Thresholding_(image_processing))
- [18] Zákon č. 361/2000 Sb., o provozu na pozemních komunikacích, § 62 a § 64.